STACY-KELVIN ADAISKY IRUTINGABO

B211200566

BİLİŞİM SİSTEMLERİ MÜHENDİSLİĞİ BÖLÜMÜ

**BE-FİT E-COMMERCE SİTESİ**

**GİRİŞ**

Spor tutkunlarının kaliteli ekipmanlara şeffaf bir erişim talep ettiği bir dönemde dijital peyzaj, tüketicileri istedikleri ekipmana bağlamak için merkezi bir rol oynar. Bu rapor, spor ekipmanlarına odaklanan dinamik ve kullanıcı dostu bir platformun oluşturulmasında kullanılan karmaşık web teknolojilerini incelemektedir. Dünya teknoloji ile sporun birleşimini kucaklarken, projemiz spor tutkunları ile tercih ettikleri ekipman arasındaki boşluğu kapatmayı amaçlamaktadır.

Bu web tabanlı girişimin amacı, kullanıcıların geniş bir spor ekipmanı yelpazesini keşfetmelerine, seçmelerine ve satın almalarına olanak tanıyan kapsamlı ve görsel olarak çekici bir çevrimiçi alan sağlamaktır. Siz yeni bir fitness yolculuğuna başlamak isteyen bir amatör veya kaliteli profesyonel ekipman arayan deneyimli bir sporcu olun, web sitemiz HTML, CSS, Flexbox, Bootstrap ve JavaScript'in sorunsuz entegrasyonu ile çeşitli ihtiyaçlara cevap vermeye çalışmaktadır.

**NEDEN BU PROJE?**

Bu projenin merkezine spor ekipmanları platformunun seçilmesi, teknoloji ve atletizme duyulan kişisel bir tutkudan kaynaklanmaktadır. E-ticaret peyzajının evrimi ve özel spor ekipmanlarına artan talep göz önüne alındığında, bu maceraya atılma kararı, geleneksel online alışverişi aşan dijital bir alan yaratma arzusundan kaynaklanmıştır.

Spor ekipmanları endüstrisi sadece profesyonellere hitap etmekle kalmaz, aynı zamanda yeni spor ve fitness aktivitelerine giriş yapan bireylere de kapılarını açar. Bu proje, teknoloji ve sporu sorunsuz bir şekilde entegre eden bir platform oluşturarak kullanıcıların çeşitli ihtiyaçlarına cevap vermeyi amaçlamaktadır, bu da tutkulu kişiler arasında bir topluluk hissi oluşturmayı teşvik eder.

**GELECEKTEKİ GELİŞİM PERSPEKTİFLERİMİZ**

Geliştirilmiş Kullanıcı Etkileşimi: Kullanıcılara daha etkileşimli ve çekici bir deneyim sunmak için gelişmiş JavaScript özelliklerini entegre etmek. Ürünlerin gerçek zamanlı özelleştirmesi, kişiselleştirilmiş öneriler ve sosyal medya entegrasyonu gibi özellikler, kullanıcı katılımını artırabilir.

Ürün Yelpazesinin Genişletilmesi: Sürekli olarak envanteri güncellemek ve daha geniş bir spor ve aktivite yelpazesini kapsamak. Tanınmış marka ve üreticilerle işbirlikleri, platformun güvenilirliğine katkıda bulunabilir ve daha geniş bir izleyici kitlesi çekebilir.

Mobil Optimizasyon: Mobil gezinmenin giderek artan yaygınlığına yanıt olarak, platformun farklı cihazlarda sorunsuz bir deneyim için optimize edilmesine öncelik vermek. Duyarlı tasarım prensiplerini uygulayarak akıllı telefonlar ve tabletler üzerinde erişilebilirlik ve işlevselliği sağlamak.

Topluluk Oluşturma: Kullanıcıların deneyimlerini, fikirlerini ve önerilerini paylaşabileceği bir topluluk alanı oluşturmak. Tartışma forumları, kullanıcı tarafından oluşturulan içerikler ve uzman blogları entegre ederek spor tutkunları topluluğu içinde bir aidiyet duygusu oluşturmak.

Yükselen Teknolojilerin Entegrasyonu: Ürünlerin sanal denemeleri için artırılmış gerçeklik (AR), kişiselleştirilmiş ürün önerileri için yapay zeka (AI) ve güvenli işlemler ve ürün kimliği doğrulama için blok zinciri gibi yükselen teknolojilerin entegrasyonunu keşfetmek.

Bu geleceğe yönelik düşüncelere dayanarak, projenin amacı sadece mevcut beklentilere cevap vermek değil, aynı zamanda e-ticaret ve dijital spor platformlarının değişen manzarasına önceden tahmin yapmak ve uyum sağlamaktır.

1. **HOME SAYFASI**

* **NAVBAR**

HTML, başlığın yapısını logo, zaman, gezinme ve simgelerle tanımlar.

CSS, sabit konum, renkler, esnek düzen vb. dahil olmak üzere başlığın stillerini tanımlar.

* **ANLIK SAAT**

İlk olarak HTML'de, içinde h1 etiketi bulunan bir div oluşturdum. Bu h1 etiketi saatin görüntüleneceği yerdir ve ona "geçerli zaman" kimliğini verdim, böylece onu JavaScript ile hedefleyebilirim.

Daha sonra JavaScript'te, tarayıcıya saati her saniyede (1000 milisaniye) güncellemesini söylemek için setInterval'ı kullandım. Bu fonksiyonun içinde Date nesnesinin geçerli zamanı temsil eden yeni bir örneğini oluşturdum. Daha sonra h1 etiketimin içeriğini bu sefer local formatta güncelledim.

* **MENÜYÜ VE ARAMA FORMÜLÜNÜ**

JavaScript, menüyü ve arama formülünü izlemek veya önbelleğe almak için "aktif" sınıfının temeli olarak gezinme çubuğu etkileşimlerini içerir.

* **Footer**

**HTML**

Altbilginin HTML kodundaki bölümlere ayrılması, <section>, <div>, <h4>, <p> ve <form> gibi HTML etiketleri kullanılarak gerçekleştirilir. Bu etiketler, altbilgi içeriğini her biri kendi amacına sahip farklı bölümlere ayırmak için kullanılır.

**CSS**

İletişim bilgilerini düzenlemek için bir Grid düzeni uyguladım.

1. **REVİEW SAYFASI**

Bootstrap dosyalarının entegrasyonu:

Aşağıdaki kodu kullanarak ana Bootstrap stil sayfasını sayfama ekledim:

**<link rel = "stylesheet" href = "https://cdn.jsdelivr.net/npm/bootstrap@5.2.0/dist/css/bootstrap.min.css">**

Bootstrap bileşenlerinin bu kodla çalışması için gereken JavaScript dosyalarını da ekledim:

**<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0/dist/js/bootstrap.bundle.min.js"></script>**

Bootstrap bileşenlerini kullanma:

**Carrousel:**

Referansları görüntülemek için bir Bootstrap **Carrousel** entegre ettim. Aşağıda ilgili kodun bir özeti verilmiştir:

Etkileşimli bir atlıkarınca Carrousel için Bootstrap ‘in **Carousel, carousel-item, carousel-control-prev** gibi önceden tanımlanmış sınıflarını kullandım. Eklediğim Bootstrap JavaScript betiği atlıkarıncanın nasıl çalıştığını yönetiyor.

**Önyükleme Simgeleri:**

Sosyal medya bağlantılarım için Bootstrap simgelerini kullanmayı seçtim. İşte bir örnek:

**<a href = "#" class = "bi bi-facebook"></a>**

Bi, bi-facebook vb. sınıflar Bootstrap Icons tarafından sağlanır. Bağlantılarıma stilize simgeler ekliyorlar.

1. **FAQ SAYFASI:**

* **Const buttons = document.querySelectorAll(button);:**

Bu kod satırı, sayfadaki tüm <button> öğelerini seçer ve bunları düğmeler adı verilen bir NodeList'te saklar.

* **buttons.forEach(button => {... });** :

Listedeki her button üzerinde yineleme yapmak için forEach döngüsü kullanılır.

* **Button.addEventListener('tıklayın', () => { ... });** :

Tıklamaları algılamak için her düğmeye bir olay dinleyicisi eklenir.

* **const FAQ = butonu.nextElementSibling;** :

Tıklama geri çağırma işlevi içinde bu satır, ilgili sorunun yanıtı olan düğmenin hemen sonraki öğesini seçer.

* **const icon = button.children[1];** :

Bu satır, yanıtın görünür mü yoksa gizli mi olduğunu belirtmek için kullanılan ok simgesi olan düğmenin ikinci alt öğesini seçer.

* **faq.classList.toggle('show');** :

Yanıt öğesine 'show' CSS sınıfını ekler veya kaldırır. Bu sınıf, maksimum yüksekliği ve opaklığı ayarlayarak yanıtı göstermek veya gizlemek için kullanılır.

* **icon.classList.toggle('rotate');** :

Ok simgesine 'döndürme' CSS sınıfını ekler veya kaldırır. Bu sınıf, yanıt görüntülendiğinde simgeyi döndürmek için kullanılır.

butona tıklandığında ilgili yanıt gösterilir veya gizlenir ve ok simgesi mevcut durumu gösterecek şekilde döner.

1. **PRODUCT VE DİSCOUNT SAYFASI**

**1. Sepetin açılması ve kapatılması:**

- Tıklandığında alışveriş sepetinin görünmesini ve kaybolmasını sağlayan "openShopping" ve "closeShopping" düğmelerini oluşturdum. Sanki açılıp kapanıyormuş gibi bir his var.

**2. Ürün listesi:**

- Her birinin kendi adı, görseli ve fiyatı olan hayali ürünlerden oluşan bir listem var. Satılacak şeylerden oluşan sanal koleksiyonum gibi.

**3. Uygulamayı başlatma:**

- İlk başta sayfa yüklendiğinde, ürün listeme bakacak ve her biri için görsel parçalar oluşturacak 'initApp' adında bir fonksiyonum var. Bu, vitrinimin ilk kurulumuna benziyor.

**4. Sepete Ekle:**

- Birisi "Sepete Ekle"ye tıkladığında, seçilen ürünü listemden alıp sepeti temsil eden özel bir alana kopyalayan "addToCard" fonksiyonunu uyguladım. Bir şeyi gerçek bir arabaya koymak gibi.

**5. Sepet güncellemesi:**

- Ayrıca her seferinde sepette kaç ürün olduğunu ve ödenecek toplam tutarı yeniden hesaplayan bir tür otomatik sistem ("reloadCard") düzenledim.

**6. Miktar değişikliği:**

- Sepetimdeki bir ürünün adedini değiştirmek istediğimde, fonksiyon çağıran (`changeQuantity`) "+"/"-" butonlarım var. Miktarı ayarlar ve toplamları günceller. Sepetimdeki ürün sayısını ayarlamak gibi.

**7. Sepet açma ve kapama etkinlikleri:**

- Ayrıca "Sepeti aç" veya "Sepeti kapat" gibi düğmelere tıkladığımda belirli eylemler de ekledim. Bu siteme etkileşimli bir his veriyor.

1. **CONTACT SAYFASI**

İletişim bölümü iki farklı bölüme ayrılmış durumda: birinci kısım (<div class="content">) başlık ve kısa bir açıklama içeriyor, ikinci kısım (<div class="container">) ise iletişim bilgilerini ve formu içeriyor.

**İletişim Bilgileri :**

İletişim bilgilerini düzenli bir şekilde göstermek için bir ızgara düzeni (<div class="contactInfo">) kullandım. Her kutu (<div class="box">) bir simge ve açıklayıcı metin içeriyor.

**İletişim Formu :**

Form (<div class="contactForm">) iletişim bilgilerinin yanına yerleştirilmiştir. İsim, e-posta ve mesaj için alanlar içerir ve bir gönder düğmesi bulunmaktadır.

**Genel Stil :**

İletişim bölümüne tutarlı bir stil uyguladım, uygun genişlikler, boşluklar ve renkler belirledim. Arka plan background: var(--primary-color) olarak ayarlanmıştır.

**Bilgi Kutularının Stili :**

Bilgi kutularının (<div class="box">) simgeleri ortalanmış ve birincil renkli bir arka plana sahiptir. Kutu başlıkları #EE1C47 renk ile vurgulanmıştır.

**Form Stili :**

Formun arka planı birincil renkli, alanlar düzenli bir şekilde yerleştirilmiş ve stilize edilmiş bir gönder düğmesi bulunmaktadır. Kullanıcı deneyimini artırmak için alanların odaklanma ve doğrulama efektleri de eklenmiştir.